

# GUIÓN INICIAL

## PMDM



## **Historia y Evolución de los Dispositivos Móviles**

El primer teléfono móvil fue inventado en 1973 por Motorola. Ese año, **el ingeniero de Motorola Martin Cooper hizo la primera llamada de teléfono móvil de la historia con el DynaTAC 8000X.**

Este pesaba 1,1 kg y era bastante grande a comparación de los modelos actuales. Ofrecía un tiempo de conversación de unos 30 minutos y necesitaba 10 horas para recargarse.

Los teléfonos móviles se popularizaron durante la revolución móvil en los 90. **En 1990, el número de usuarios de móviles era de 11 millones, en 2020, ronda los 2.500 millones.**

El primer smartphone se presentó en 1994, **el Simon de IBM fue el primer dispositivo con aplicaciones y pantalla táctil, por lo que se considera el primer smartphone del mundo.**

En 2007 llegó el **iPhone por parte de Apple.** Conocido como **iPhone 2G, salió al mercado sorpendiendo a todos** gracias a la eliminación de la mayoría de los botones físicos, dando lugar a una interfaz táctil. Permitía por primera vez navegar por Internet en un dispositivo relativamente pequeño. Asimismo, **fue el primer móvil de la historia en incluir aplicaciones dedicadas para YouTube y Google Maps.**

Este abrumador éxito **hizo que Google fuese forzado a doblar la apuesta de Apple.** Android 1.0 debutó en el HTC G1 en 2008 y Google se aseguró de que incluyera una gran pantalla táctil, un navegador web y la tienda de aplicaciones Android Market.

**Dos años más tarde Samsung también lanzó el primer smartphone 4G: el Samsung SCH-R900. También se lanzó el Nexus S, el primer smartphone Android con NFC.**

Por supuesto, a partir de aquí las evoluciones en el mercado se han ido poco a poco ralentización y cada vez resulta más complejo innovar en el sector móvil. Muchas empresas buscan esa diferencia con cargas rápidas, nuevas funciones, la conexión saltelital tan novedosa, etc.

Fuente : <https://computerhoy.com/moviles/historia-telefonos-moviles-origen-actualidad-1181484>

## **Características de los Dispositivos Móviles**

- Son aparatos pequeños: la mayoría se pueden transportar en el bolsillo del propietario o en un pequeño bolso.
- Tienen capacidad de procesamiento.
- Tienen conexión permanente o intermitente a una red.
- Tienen memoria (*RAM*, tarjetas *MicroSD*, *flash*, etc.).
- Normalmente, se asocian al uso individual de una persona, tanto en posesión como en operación, la cual generalmente puede adaptarlos a su gusto.
- Tienen una alta capacidad de interacción mediante la pantalla o el teclado.

Un dispositivo móvil puede definirse con cuatro características que lo diferencian de otros dispositivos :

1. **Movilidad.** La característica más evidente de un dispositivo móvil. Se entiende por movilidad la cualidad de un dispositivo para ser transportado o movido con frecuencia y facilidad. Los dispositivos móviles son aquéllos que son lo suficientemente pequeños como para ser transportados y utilizados durante su transporte.
2. **Reducido tamaño.** Se entiende por tamaño reducido la cualidad de un dispositivo móvil de ser fácilmente usado con una o dos manos sin necesidad de ninguna ayuda o soporte externo. El tamaño reducido también permite transportar el dispositivo cómodamente por parte de una persona.
3. **Capacidad de comunicación inalámbrica.** Otro concepto importante es el término inalámbrico. Por comunicación inalámbrica se entiende la capacidad que tiene un dispositivo de enviar o recibir datos sin la necesidad de un enlace cableado. Por lo tanto, un dispositivo inalámbrico es aquél capaz de comunicarse o de acceder a una red sin cables.
4. **Capacidad de interacción con las personas.** Se entiende por interacción el proceso de uso que establece un usuario con un dispositivo. Entre otros factores, en el diseño de la interacción intervienen disciplinas como la usabilidad y la ergonomía.

Fuente: <https://www.ceupe.com/blog/que-son-los-dispositivos-moviles.html>

### **Limitaciones de las aplicaciones desarrolladas para Dispositivos Móviles**

- Las pantallas son más pequeñas y la resolución es menor.
- Los procesadores son menos potentes.
- La RAM es más limitada.
- La tasa de transferencia de datos es menor.
- Las conexiones de datos suelen ser menos estables y el coste de transferencia de estos más alto.
- Exigen constantes actualizaciones
- Hay que distribuirlas a través de distintas tiendas.
- Ocupan un espacio que en este tipo de dispositivos suele ser más limitado.

Fuentes : <https://www.edgardandrea.com/uso-de-la-aplicacion-en-un-dispositivo-real/>

<https://gunkastudios.com/ventajas-y-desventajas-de-una-aplicacion-movil/>

### **Principales S.O.**

- **Apple iOS:** El sistema operativo **Apple iOS** es el usado en todos los móviles **iPhone**, además de versiones antiguas de **iPad** (ahora usan la variante **iPad OS**) y los **iPod Touch**. Es un **sistema de código cerrado**. No se conoce el código de iOS. No puede ser mejorado ni auditado por la comunidad
- **Android:** **Android es un sistema operativo de Google** que es **mayormente de código abierto**. Esto, junto con el extendido uso del sistema operativo, ha llevado a que los fabricantes **creen sus propias modificaciones de Android** para intentar diferenciarse de la competencia. Es lo que conocemos como **ROM personalizadas** o **capas de personalización**.

Fuente: <https://www.profesionalreview.com/2022/08/06/sistemas-operativos-moviles/>

## Tecnología para el desarrollo de aplicaciones móviles

### iOS con Swift:

Es el lenguaje de programación ofrecido por Apple para el desarrollo de aplicaciones que **funcionen en sus plataformas: iOS, macOS, tvOS, watchOS y iPadOS.**

### Android con Kotlin:

Este lenguaje permite crear apps nativas de alta calidad y gran rendimiento **en todas las plataformas soportadas por Android:** smartphones, televisores, relojes, etc. Su principal desventaja es la misma que las aplicaciones desarrolladas con Swift. Duplicidad de esfuerzos, equipo y código.

### Ionic:

El framework probablemente más popular que nos permite desarrollar aplicaciones multiplataforma. Nos ayuda a **construir apps con HTML, CSS3 y Javascript** empleando todo el potencial de las tecnologías de desarrollo web.

### Flutter:

Una herramienta *desarrollada por Google* para el desarrollo de aplicaciones multiplataforma. Está **basada en Dart** y aunque se presentó en 2015, no se lanzó hasta 2018.

Una gran ventaja frente a otras soluciones es que la aplicación desarrollada puede **funcionar hasta en 4 plataformas: iOS, Android, Web y Desktop.**

### React Native:

Es un framework **creado por Facebook en 2015** para el desarrollo de apps multiplataforma.

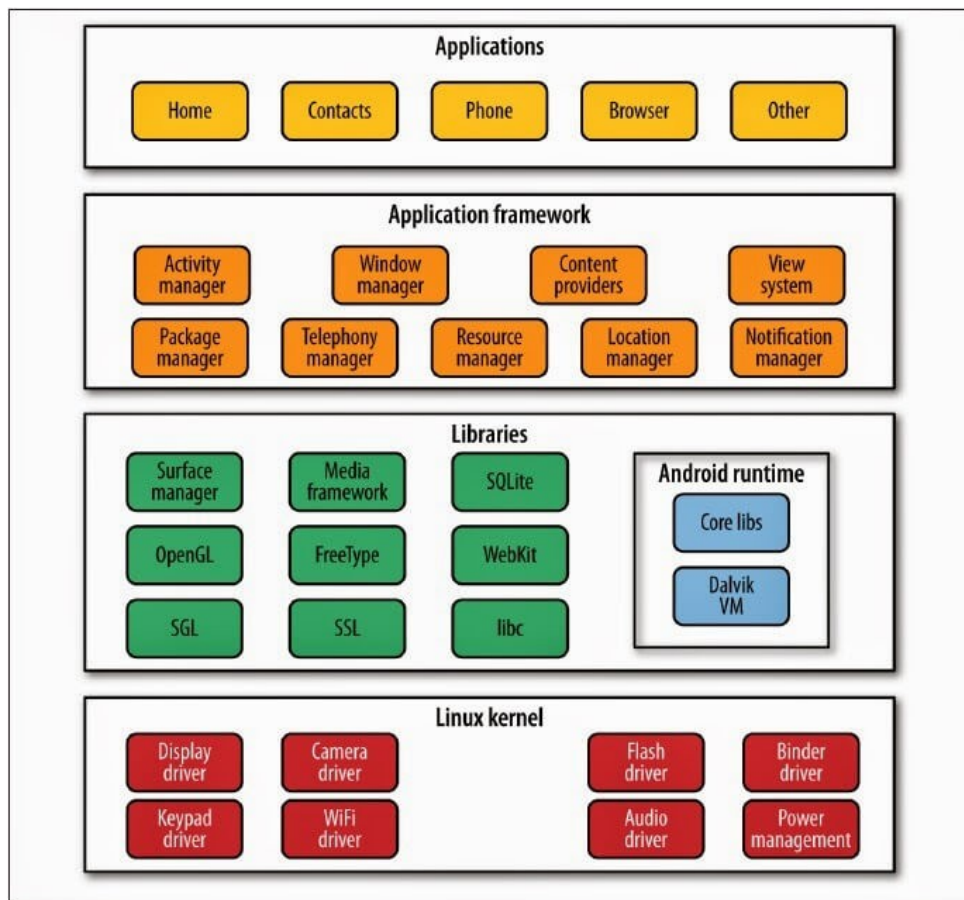
Al igual que Ionic, también **utiliza Javascript**, pero lo hace bajo el pretexto de orquestar controles de interfaz de usuario (UI) específicos de la plataforma.

### Xamarin:

Es una plataforma creada por los desarrolladores de Mono y posteriormente **adquirida por Microsoft**. Utiliza C# y al igual que los otros frameworks consigue crear aplicaciones nativas con muy buen rendimiento. La interfaz de usuarios requiere un mayor tiempo de construcción y la comunidad no es tan activa.

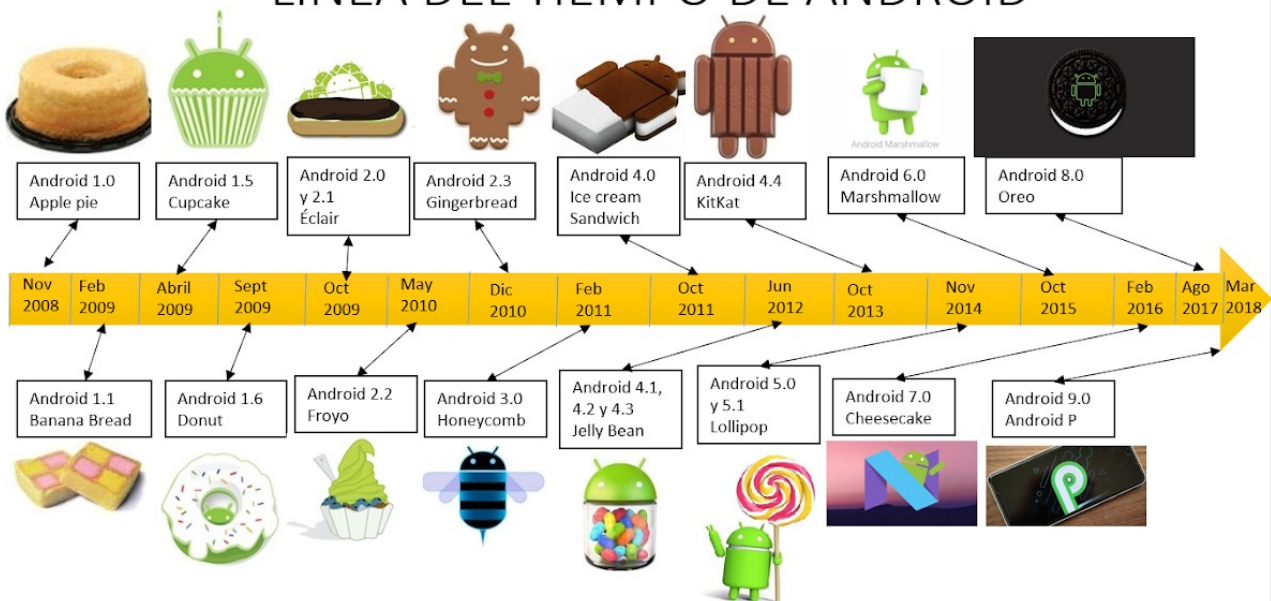
Fuente: <https://www.armadilloamarillo.com/blog/mejores-tecnologias-para-desarrollo-de-aplicaciones-moviles/>

## Arquitectura Android



## Versiones

### LINEA DEL TIEMPO DE ANDROID



## **Entorno integrado de trabajo**

Android Studio es el entorno de desarrollo integrado (IDE) oficial que se usa en el desarrollo de apps para Android. Basado en el potente editor de código y las herramientas para desarrolladores de IntelliJ IDEA , Android Studio ofrece aún más funciones que mejoran tu productividad cuando compilas apps para Android, como las siguientes:

- Un sistema de compilación flexible basado en Gradle
- Un emulador rápido y cargado de funciones
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android
- Ediciones en vivo para actualizar elementos componibles en emuladores y dispositivos físicos, en tiempo real
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra
- Variedad de marcos de trabajo y herramientas de prueba
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros
- Compatibilidad con C++ y NDK
- Compatibilidad integrada con Google Cloud Plataform que facilita la integración con Google Cloud Messaging y App Engine

Fuente: <https://developer.android.com/studio/intro?hl=es-419>